REMARKS

Claims 1-4, 9-19, 71, 74-81, 86, 88, 92, and 94-111 were pending and stand rejected. Claim 92 has been cancelled. Claims 1, 71, 86, 88, and 94 have been amended. Claim 112 has been added. Claims 1-4, 9-19, 71, 74-81, 86, 88, and 94-112 are pending upon entry of this amendment.

Claims 1, 4, 9-10, 12-18, 71, 74-75, 77-78, 94-96, 98-104, and 106-107 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Grinstein. Applicants respectfully traverse.

CLAIM 1

As amended, claim 1 recites:

> In a computer-implemented animation system, a method for animating an object, the method comprising:
>
> receiving a first input, the first input specifying a first parameter behavior, the first parameter behavior indicating how to change a value of a first parameter over time, wherein the first parameter applies to one element of a group consisting of a filter applied to the object and a generator applied to the object, and wherein the filter comprises an image processing effect, and wherein the generator comprises a repeating image;
>
> animating the object by changing the value of the first parameter over time according to the specified parameter behavior; and
>
> outputting the animated object.

As recited in claim 1, a "parameter behavior" indicates how to change, over time, a value of a parameter of a filter or generator that is applied to the object. A filter comprises "an image processing effect" (¶¶245, 841, 1627), and a generator comprises "a repeating image" (¶1567). A filter or generator can be customized using a parameter. The value of the parameter affects the filter or generator, which in turn affects the appearance of an object. For example, a filter with a parameter value of 1 will result in a different appearance than the same filter with a parameter value of 10. The value of a filter's parameter or a generator's parameter can be programmatically animated (i.e., changed over time) by using a parameter behavior (¶¶245, 248, 487). This results in different appearances as time goes on, based on the different values of the parameter.

Grinstein discusses modeling motion in computer applications (title). Grinstein does not disclose, teach, or suggest the claimed element "the first parameter behavior indicating how to change a value of a first parameter over time, wherein the first parameter applies to one element of a group consisting of <u>a filter applied to the object</u> and a generator applied to the object, and wherein <u>the filter comprises an image processing effect</u>, and wherein the generator comprises a repeating image" (emphasis added).

Applicants agree with the Examiner that Grinstein does not use the language "filters" (Detailed Action, page 3). The Examiner argues that the Ramp and Ease controls applied to the Swing motion applied to the object are analogous to filters (Detailed Action, p. 3). Applicants disagree.

The Ramp sliders control the acceleration and deceleration in and out of an entire motion (51:24-25). The Ease sliders control the acceleration and deceleration to and from a displacement sequence (51:25-27). As recited in claim 1, <u>a filter comprises an image processing effect</u>. Neither the Ramp controls nor the Ease controls are image processing effects. Thus, neither the Ramp controls nor the Ease controls are filters.

Grinstein does not mention generators.

Thus, Grinstein does not disclose, teach, or suggest the claimed element "the first parameter behavior indicating how to change a value of a first parameter over time, wherein the first parameter applies to one element of a group consisting of a filter applied to the object and a generator applied to the object, and wherein the filter comprises an image processing effect, and wherein the generator comprises a repeating image."

Therefore, claim 1 (as amended) is patentable over Grinstein.

<u>CLAIMS 71, 94</u>

As amended, claim 71 recites:

A method for animating an object using a behavior, comprising:

outputting an original animation for the object according to a first parameter behavior, the first parameter behavior indicating how to change a value of a first parameter over time, wherein the first parameter applies to a motion behavior applied to the object;

concurrently with outputting the original animation:

receiving a first user input, the first user input directly specifying a second parameter of the motion behavior; and

receiving a second user input, the second user input directly specifying a second parameter behavior, the second parameter behavior indicating how to change a value of the second parameter over time; and

outputting an updated animation for the object according to the first parameter behavior and further according to the second parameter behavior.

As recited in claim 71, a "parameter behavior" indicates how to change, over time, a value of a parameter of a "motion behavior." As explained in the application, in one embodiment, a motion behavior changes an object's position over time, thereby animating the object (¶247). A motion behavior can be customized using a parameter (¶9). The value of the parameter affects the motion behavior, which in turn affects the animation of an object. For example, a motion behavior with a parameter value of 1 will result in a different animation than the same motion behavior with a parameter value of 10.

The value of a motion behavior's parameter can be programmatically animated (i.e., changed over time) by using a parameter behavior (¶402). This results in different animations as time goes on, based on the different values of the parameter. For example, consider the Drag parameter of the Orbit Around motion behavior (¶404). If the value of the Drag parameter is kept constant over time, the object moves in a regular orbit with a circular motion path (¶404; FIG. 34). If, instead, the value of the Drag parameter is increased over time (e.g., using the Ramp parameter behavior), the object's orbit slowly decays over time, causing the object to fall towards the center of the orbit with a spiral motion path (¶404; FIG. 35).

Grinstein does not disclose, teach, or suggest the claimed elements "receiving a first user input, the first user input directly specifying a second parameter of the motion behavior" and "receiving a second user input, the second user input directly specifying a second parameter behavior, the second parameter behavior indicating how to change a value of the second parameter over time." Note that this portion of claim 71 recites receiving <u>two user inputs</u>: a) a first user input that <u>directly specifies</u> a second parameter of the motion behavior and b) a second user input that <u>directly specifies</u> a second parameter behavior, the second parameter behavior indicating how to change a value of the second parameter over time.

The Examiner argues that Grinstein's controllers correspond to this claimed element. Controllers, such as the Sway controller (52:47-56; FIGS. 31-32) and the Wind controller (52:57-61; FIGS. 29-30), are used to set the parameters of various "controlled motions" (49:59-50:15). Specifically, the variable parameters of a controlled motion are derived from the controllers (49:61-62).

Assume, *arguendo*, that the claimed element "motion behavior" corresponds to a controlled motion (a type of pre-defined animation) in Grinstein. The controller dialog boxes in FIGS. 29 and 32 enable a user to set a value for a <u>controller</u> parameter (e.g., minimum angle or strength), which is then used to <u>derive</u> a parameter of a <u>controlled motion</u>. Claim 71 recites "receiving a first user input, the first user input <u>directly specifying a second parameter of the motion behavior</u>." The controller dialog boxes do not enable a user to <u>directly specify a parameter of a controlled motion</u> (e.g., the parameter that is supposed to be affected by the controller). Instead, the controller dialog boxes <u>indirectly influence</u> the controlled motion (49:60-61). Thus, this portion of Grinstein does not disclose, teach, or suggest the claimed element "receiving a first user input, the first user input directly specifying a second parameter of the motion behavior."

Claim 71 also recites "receiving a second user input, the second user input directly specifying a second parameter behavior, the second parameter behavior indicating how to change a value of the second parameter over time." The controller dialog boxes do not enable a user to directly specify how to change a value of a parameter of a controlled motion. Instead, the controller dialog boxes indirectly influence the controlled motion (49:60-61). Thus, this portion of Grinstein does not disclose, teach, or suggest the claimed element "receiving a second user input, the second user input directly specifying a second parameter behavior, the second parameter behavior indicating how to change a value of the second parameter over time."

Thus, Grinstein does not disclose, teach, or suggest the claimed elements "receiving a first user input, the first user input directly specifying a second parameter of the motion behavior" and "receiving a second user input, the second user input directly specifying a second parameter behavior, the second parameter behavior indicating how to change a value of the second parameter over time."

Therefore, claim 71 (as amended) is patentable over Grinstein. Claim 94 (as amended) recites similar language is also patentable over Grinstein for at least the same reasons.

CLAIM 88

Claims 11, 88, 97, and 110 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Grinstein. Applicants respectfully traverse.

On October 22, 2007, the Examiner and the undersigned attorney had a telephone conversation during which they discussed claim 88 as previously pending and Grinstein. No agreement was reached. The substance of that conversation is set forth herein.

As amended, claim 88 recites:

In a computer-implemented animation system, a method for animating an object, the method comprising:

receiving an input, the input specifying a behavior to apply to the object, the behavior indicating how to change a value of a parameter of the object over time;

animating the object by changing the value of the parameter of the object over time according to the specified behavior; and

outputting the animated object;

wherein the behavior comprises one from a group consisting of:

a Drag behavior, which changes a position of the object based on a simulated friction regardless of the object's proximity to another object; and

a Rotational Drag behavior, which changes a rotation of the object based on a simulated friction regardless of the object's proximity to another object.

Claim 88 recites, in part, "a behavior to apply to the object … wherein the behavior comprises one from a group consisting of: a Drag behavior, which changes a position of the object based on a simulated friction <u>regardless of the object's proximity to another object</u>; and a Rotational Drag behavior, which changes a rotation of the object based on a simulated friction <u>regardless of the object's proximity to another object</u>" (emphasis added). As described in the pending application, the Drag behavior is meant to be applied to a moving object (i.e., an object whose position parameter is changing over time) (¶623). The Rotational Drag behavior is meant to be applied to a spinning object (i.e., an object whose rotation parameter is changing over time) (¶731). These behaviors can be used, for example, to simulate the effect of friction on a moving object (¶¶623, 731). This friction can be thought of as the difference between an object moving through a gas (e.g., air) and the same object moving through a liquid (e.g., water). In other words, this friction affects the moving object whether or not the moving object is touching and/or close to another object.

Grinstein does not disclose, teach, or suggest the claimed element "a behavior to apply to the object … wherein the behavior comprises one from a group consisting of: a Drag behavior, which changes a position of the object based on a simulated friction regardless of the object's proximity to another object; and a Rotational Drag behavior, which changes a rotation of the

object based on a simulated friction regardless of the object's proximity to another object." A boundary behavior's "gain" and "bias" parameters, which were cited by the Examiner, can be used to simulate effects of gain or loss of momentum (e.g., due to friction) (36:17-20). However, even if the object changes momentum, it must still move according to a boundary behavior. Boundary behaviors, which include reflect, clamp, and onto, require that the object be close to another object's boundary (35:25-36:15). Thus, friction can be simulated only if a second object is present. Friction cannot be simulated for an object if the second object is absent. It follows that Grinstein does not change a position of an object based on a simulated friction regardless of the object's proximity to another object.

As mentioned above with respect to claim 71, Grinstein uses a Controller to set the parameters of various controlled motions. Grinstein mentions a Viscosity Controller that has various parameters (strength, location, shape, and focus) (53:2-4). However, Grinstein does not explain what the Viscosity Controller is used for or which controlled motion is controlled by the Viscosity Controller (53:1-6).

Thus, Grinstein does not disclose, teach, or suggest the claimed element "a behavior to apply to the object … wherein the behavior comprises one from a group consisting of: a Drag behavior, which changes a position of the object based on a simulated friction regardless of the object's proximity to another object; and a Rotational Drag behavior, which changes a rotation of the object based on a simulated friction regardless of the object's proximity to another object."

Therefore, claim 88 (as amended) is patentable over Grinstein.

CLAIM 86

Claim 86 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Grinstein in view of Unuma. Applicants respectfully traverse. Additionally, for the record, Applicants traverse the Examiner's assertions concerning the motivation to combine Grinstein and Unuma.

As amended, claim 86 recites:

> In a computer-implemented animation system, a method for animating an object, the method comprising:
>
> receiving an input, the input specifying a behavior, the behavior indicating how to change a value of a parameter of the object over time;
>
> animating the object by changing the value of the parameter of the object over time according to the specified behavior; and
>
> outputting the animated object;
>
> wherein the behavior comprises an Align to Motion behavior, which changes a rotation of the object based on a motion path of the object such that the rotation is not changed if the motion path is straight, and which can be configured regarding at least one of how quickly the object's rotation changes based on a change in the object's motion path and whether or not the object's change in rotation overshoots a new direction of the object.

Claim 86 recites, in part, "wherein the behavior comprises an Align to Motion behavior, which changes a rotation of the object based on a motion path of the object such that the rotation is not changed if the motion path is straight, and which can be configured regarding at least one of how quickly the object's rotation changes based on a change in the object's motion path and whether or not the object's change in rotation overshoots a new direction of the object." As described in the pending application, the Align to Motion behavior is meant to be applied to a moving object (i.e., an object whose position parameter is changing over time) (¶581). This behavior changes the rotation of the object to match changes made to the object's direction along its motion path (¶581). The Align to Motion behavior can be used, for example, to cause an object to face the direction in which it is moving. Unlike the Snap Alignment to Motion behavior, which produces absolute changes in rotation that precisely match changes in direction, the Align to Motion behavior has a springy effect (¶582) due to the Spring Tension parameter (¶587) and the Drag parameter (¶588).

Applicants agree with the Examiner that Grinstein does not disclose, teach, or suggest the claimed element "wherein the behavior comprises an Align to Motion behavior, which changes a rotation of the object based on a motion path of the object such that the rotation is not changed if the motion path is straight." It follows that Grinstein also does not disclose, teach, or suggest the claimed element "wherein the behavior comprises an Align to Motion behavior, which changes a rotation of the object based on a motion path of the object such that the rotation is not changed if the motion path is straight, and which can be configured regarding at least one of how quickly the object's rotation changes based on a change in the object's motion path and whether or not the object's change in rotation overshoots a new direction of the object."

Unuma does not remedy this deficiency. Unuma discusses a transit point specifying unit and a moving direction controller (¶133; FIG. 17). The transit point specifying unit specifies transit points that are connected to each other with a curve so as to create a moving route designated by a position and a curve (¶133; FIG. 18). Then, the moving direction controller rotates the object so that the front side of the object is oriented to a direction of a tangent of the curve at a position of the object moving on the curve (¶133).

Claim 86 recites, in part, "which can be <u>configured</u> regarding at least one of how quickly the object's rotation changes based on a change in the object's motion path and whether or not the object's change in rotation overshoots a new direction of the object" (emphasis added). In Unuma, the object's rotation is always equal to the tangent of the object's motion path. Thus, the rotation cannot be modified in any way, let alone configured regarding at least one of how quickly the object's rotation changes based on a change in the object's motion path and whether or not the object's change in rotation overshoots a new direction of the object. It follows that Unuma does not disclose, teach, or suggest "which can be configured regarding at least one of

how quickly the object's rotation changes based on a change in the object's motion path and whether or not the object's change in rotation overshoots a new direction of the object."

Thus, neither Grinstein nor Unuma, alone or in combination, discloses, teaches, or suggests the claimed element "wherein the behavior comprises an Align to Motion behavior, which changes a rotation of the object based on a motion path of the object such that the rotation is not changed if the motion path is straight, and which can be configured regarding at least one of how quickly the object's rotation changes based on a change in the object's motion path and whether or not the object's change in rotation overshoots a new direction of the object."

Therefore, claim 86 (as amended) is patentable over Grinstein and Unuma, alone and in combination.

CLAIM 92

Claim 92 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Miller in view of Altman further in view of Land. Applicants respectfully traverse. Additionally, for the record, Applicants traverse the Examiner's assertions concerning the motivation to combine Miller, Altman, and Land. Claim 92 has been cancelled.

CLAIM 112

Although new claim 112 has not been rejected, Applicants note the following:

Claim 112 is similar to claim 87, which was cancelled in the amendment filed September 27, 2007. Claim 87 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Grinstein in view of Perlin. Claim 112 recites:

In a computer-implemented animation system, a method for animating an object, the method comprising:

receiving an input, the input specifying a behavior, the behavior indicating how to change a value of a parameter of the object over time;

animating the object by changing the value of the parameter of the object over time according to the specified behavior; and

outputting the animated object;

wherein the behavior comprises one from a group consisting of:

an Attracted To behavior, which changes a position of the object based on a position of a second object while not affecting the position of the second object;

an Attractor behavior, which changes a position of a second object based on a position of the object while not affecting the position of the object;

a Drift Attracted To behavior, which changes a position of the object based on a position of a second object while not affecting the position of the second object; and

a Drift Attractor behavior, which changes a position of a second object based on a position of the object while not affecting the position of the object; and

wherein the behavior can be modified using at least one of:

a falloff rate parameter, which determines a rate of acceleration with which an attracted object moves towards an object of attraction;

a drag parameter, which determines whether an attracted object overshoots an object of attraction; and

an influence parameter, which determines an area of influence, the area of influence determining whether an object is affected by the behavior.

Applicants agree with the Examiner that Grinstein does not disclose, teach, or suggest "an Attracted To behavior, which changes a position of the object based on a position of a second object while not affecting the position of the second object" (emphasis added). Assume, *arguendo*, that Grinstein and Perlin together disclose, teach, or suggest "wherein the behavior comprises one from a group consisting of: an Attracted To behavior…; an Attractor behavior…; a Drift Attracted To behavior…; and a Drift Attractor behavior…."

Grinstein discusses a Gravity Controller and an Attraction Controller (53:1-6). Although each Controller is described as having various parameters (strength, direction, location, shape, and focus), Grinstein does not explain what each of these parameters does (53:1-6). Specifically, Grinstein does not disclose, teach, or suggest a falloff rate parameter, drag parameter, or influence parameter, as recited in claim 112. It follows that Grinstein does not disclose, teach, or suggest the claimed element "wherein the behavior can be modified using at least one of: a falloff rate parameter…; a drag parameter…; and an influence parameter…."

Perlin does not remedy this deficiency. Perlin discusses placing an "attractor field" around an opening such as a doorway (8:20-21). Vector magnitude increases as a character nears an object (8:21-22). Perlin does not disclose, teach, or suggest <u>modifying</u> the attractor field by using a falloff rate parameter, drag parameter, or influence parameter, as recited in claim 112. It follows that Perlin does not disclose, teach, or suggest the claimed element "wherein the behavior can be modified using at least one of: a falloff rate parameter…; a drag parameter…; and an influence parameter…."

Thus, neither Grinstein nor Perlin, alone or in combination, discloses, teaches, or suggests the claimed element "wherein the behavior can be modified using at least one of: a falloff rate parameter…; a drag parameter…; and an influence parameter…."

Therefore, claim 112 is patentable over Grinstein and Perlin, alone and in combination.

OTHER CLAIMS

Claims 2, 108, and 111 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Grinstein in view of Walton. Claims 3 and 105 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Grinstein in view of Gagne. Claims 19 and 109 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Grinstein in view of Anderson. Claims 76 and 79 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Grinstein in view of French. Claims 80-81 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Grinstein in view of Sowizral.

Applicants respectfully traverse. The claims not specifically mentioned above depend from their respective base claims, which were shown to be patentable over Grinstein. In addition, these claims recite other features not included in their respective base claims. Thus, these claims

are patentable for at least the reasons discussed above, as well as for the elements that they individually recite.

Applicants respectfully submit that the pending claims are allowable over the cited art of record and request that the Examiner allow this case. The Examiner is invited to contact the undersigned in order to advance the prosecution of this application.

Respectfully submitted,
GREGORY E. NILES, ET AL.

Dated: October 29, 2007

By: /Sabra-Anne R. Truesdale/

Sabra-Anne R. Truesdale, Reg. No.: 55,687
Fenwick & West LLP
Silicon Valley Center
801 California Street
Mountain View, CA 94041
Tel.: (650) 335-7187
Fax.: (650) 938-5200